CSE 484 (Winter 2008)

Web Security

Tadayoshi Kohno

Thanks to Dan Boneh, Dieter Gollmann, John Manferdelli, John Mitchell, Vitaly Shmatikov, Bennet Yee, and many others for sample slides and materials ...

IT: Solving Obama's BlackBerry Dilemma

Posted by kdawson on Tuesday January 13, @05:56PM
from the **first-personal-communicator** dept.

CurtMonash writes

> "Much is being made of the deliberations as to whether President Obama will be able to keep using his beloved "BarackBerry." As the NYTimes details, there are two major sets of objections: infosecurity and legal/records retention. Deven Coldeway of CrunchGear does a good job of showing that the technological infosecurity problems can be solved. And as I've noted elsewhere, the 'Omigod, he left his Blackberry behind at dinner' issue is absurd. Presidents are surrounded by attendants, Secret Service and otherwise. Somebody just has to be given the j device. As for the legal c writing that will likely be surely depends on the su not? Anything he'd write Secretary of Defense? T

Read More | 271 comment

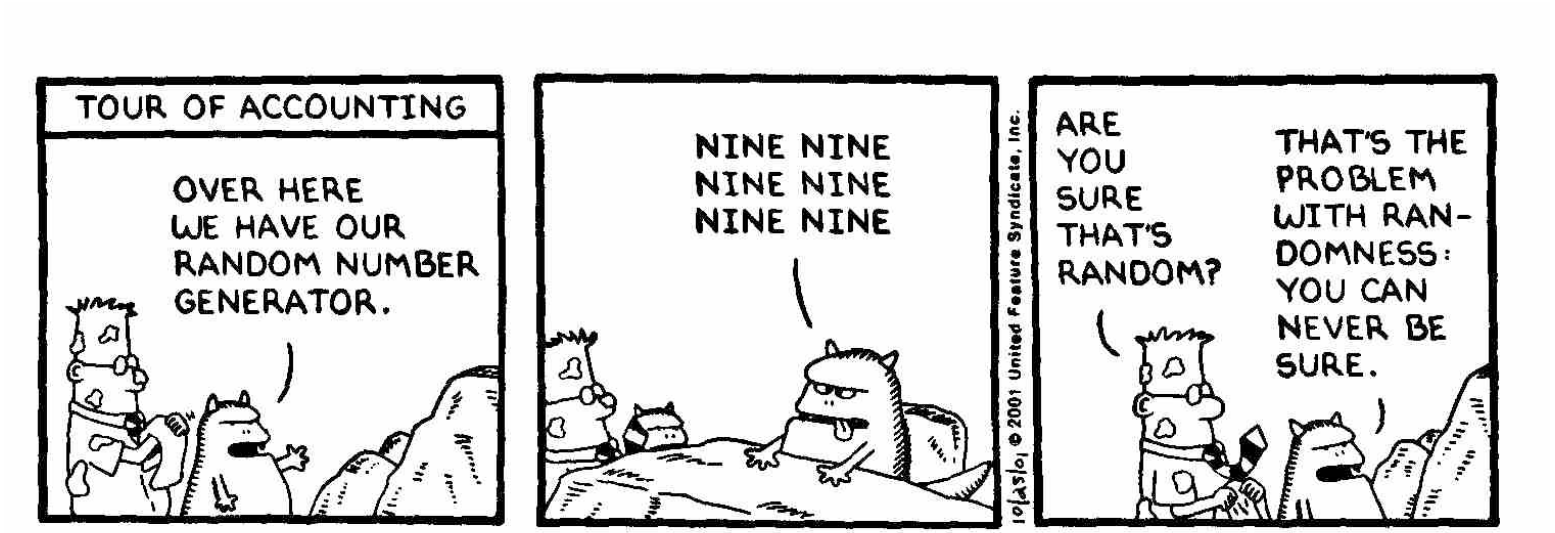
IT: Taxpayer Data At IRS Remains Vulnerable

Posted by kdawson on Tuesday January 13, @09:53PM
from the **do-as-I-say** dept.

CWmike writes

> "A new Government Accountability Office report (PDF) finds that taxpayer and other sensitive data continues to remain dangerously underprotected at the IRS. The news comes less than three months after the Treasury Inspector General for Tax Administration reported that there were major security vulnerabilities in two crucial IRS systems. Two big standouts in the latest finding: The IRS still does not always enforce strong password management rules for identifying and authenticating users of its systems, nor does it encrypt certain types of sensitive data, the GAO said."

Read More | 33 comments ▶ usa government areyousurprised irs it security story



NSA Crypto Kids:

http://www.nsa.gov/KIDS/

Industry Security?

Attack Scenarios for Encryption

- Ciphertext-Only
- Known Plaintext
- Chosen Plaintext
- Chosen Ciphertext (and Chosen Plaintext)

Attack Scenarios for Integrity

- What do you think these scenarios should be?

Birthday attacks

- Are there two people in the first 1/3 of this classroom that have the same birthday?
  - Yes?
  - No?
  - Experiment

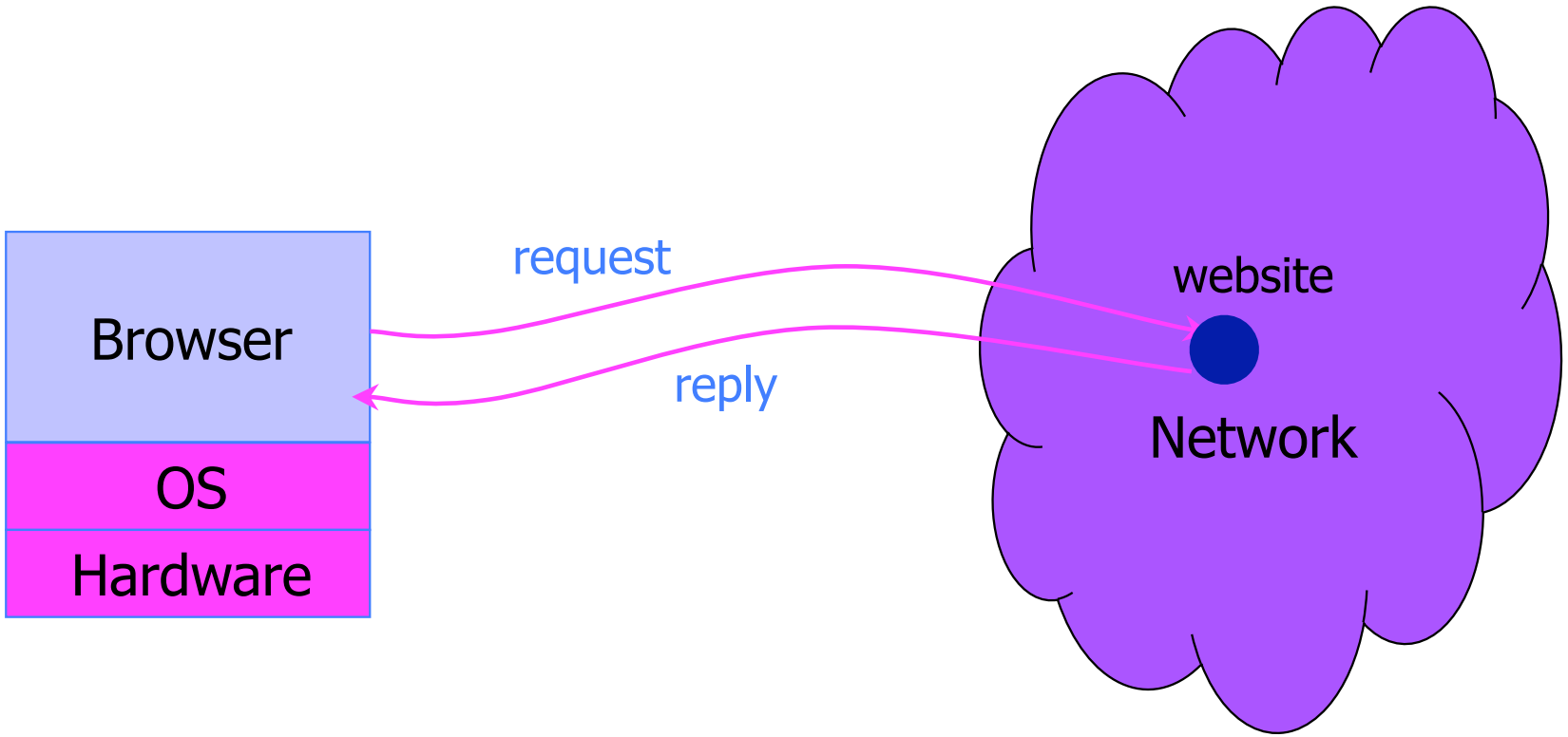
Birthday attacks

- Why is this important for cryptography?
  - 365 days in a year (366 some years)
    - Pick one person. To find another person with same birthday would take on the order of 365/2 = 182.5 people
    - Expect "collision" -- two people with same birthday -- with a room of only 23 people
  - 2^{128} different 128-bit keys
    - Pick one key at random. To exhaustively search for this key requires trying on average 2^{127} keys.
    - Expect a "collision" after selecting approximately 2^{64} random keys.
    - 64 bits of security against collision attacks, not 128 bits.

Goals for Today

- Web security
- Key issues
  - Browser is the new OS
  - State on client
  - Integrity (e.g., for pricing)
  - Privacy (e.g., cookies)
  - Website isolation (e.g., cross-site scripting)

Browser and Network



INTERNETWEEK.com
CONNECTING THE ENTERPRISE, CUSTOMERS & SUPPLIERS

February 12, 2002

Microsoft Issues New IE Browser Security Patch

By Richard Karpinski

- Microsoft has released a security patch that closes some major holes in its Internet Explorer browser
- The so-called "cumulative patch" fixes six different IE problems
- Affected browsers include Internet Explorer 5.01, 5.5 and 6.0
- Microsoft rated the potential security breaches as "critical"

Fixed by the February 2002 Patch

- Buffer overrun associated with an HTML directive
  - Could be used by hackers to run malicious code on a user's system
- Scripting vulnerability
  - Lets an attacker read files on a user's system
- Vulnerability related to the display of file names
  - Hackers could misrepresent the name of a file and trick a user into downloading an unsafe file
- … and many more

On April 13, 2004, MS announced 20 new vulnerabilities

January 7, 2007

Microsoft Security Bulletin MS07-004

A remote code execution vulnerability exists in the Vector Markup Language (VML) implementation in Microsoft Windows. An attacker could exploit the vulnerability by constructing a specially crafted Web page or HTML e-mail that could potentially allow remote code execution if a user visited the Web page or viewed the message. An attacker who successfully exploited this vulnerability could take complete control of an affected system.

Maximum Severity Rating: Critical

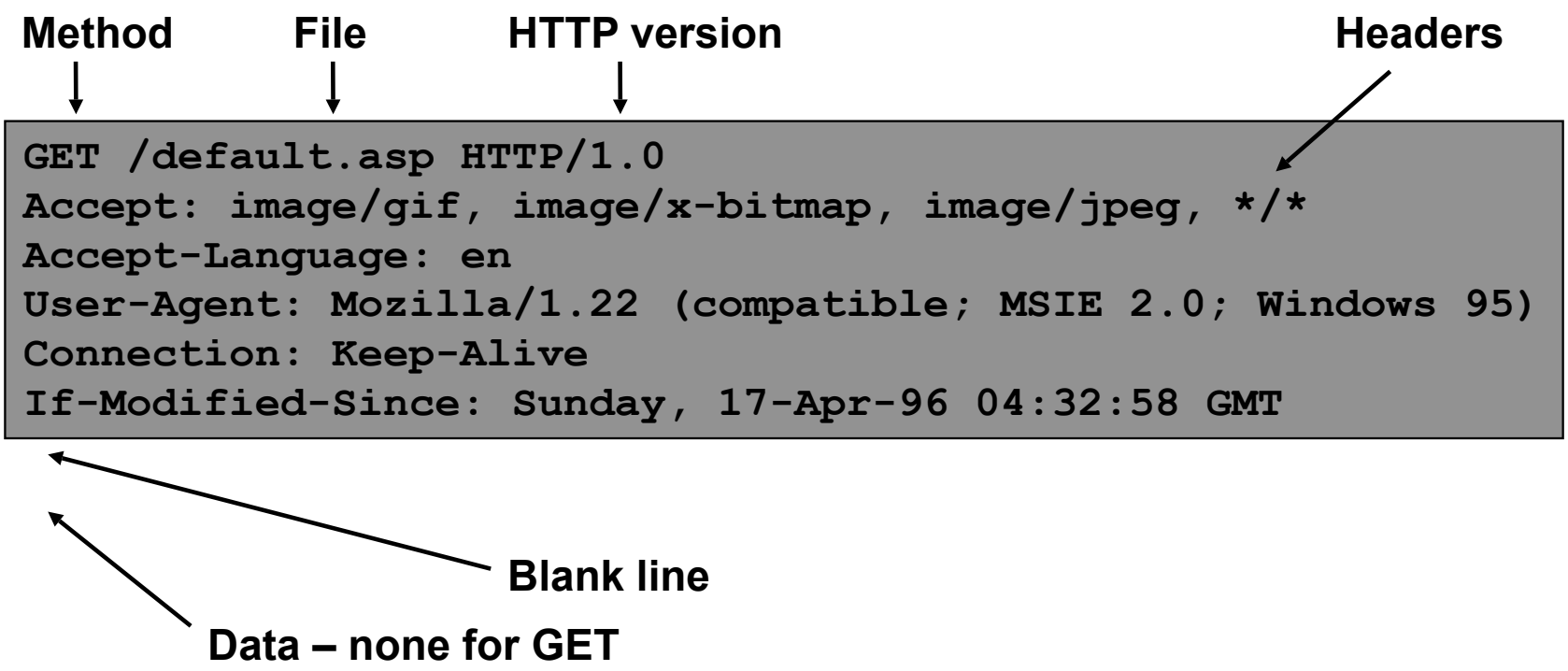
Recommendation: Customers should apply the update immediately

Browsers are becoming "mini operating systems" - complex, running third-party code, etc.

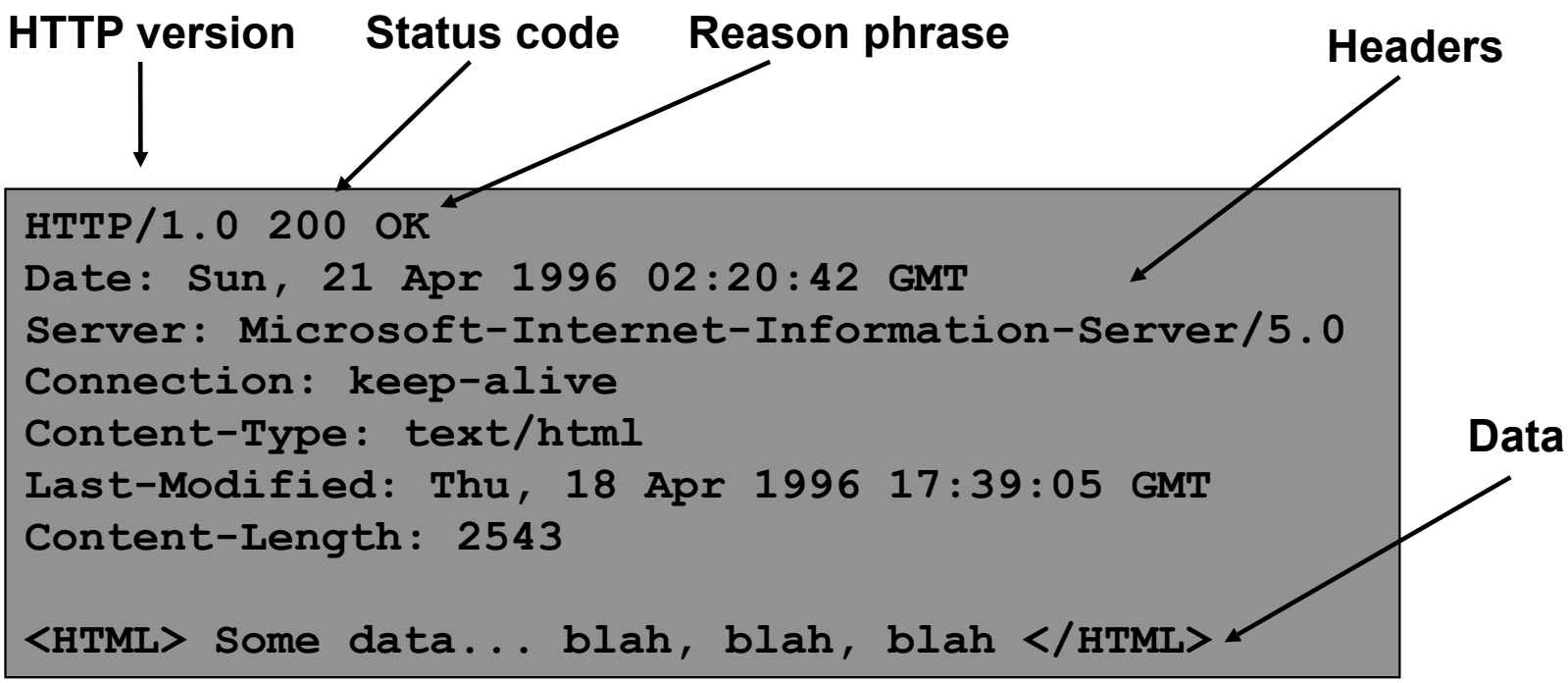
HTTP: HyperText Transfer Protocol

- Used to request and return data
  - Methods: GET, POST, HEAD, …
- Stateless request/response protocol
  - Each request is independent of previous requests
  - Statelessness has a significant impact on design and implementation of applications
- Evolution
  - HTTP 1.0: simple
  - HTTP 1.1: more complex
  - … Still evolving …

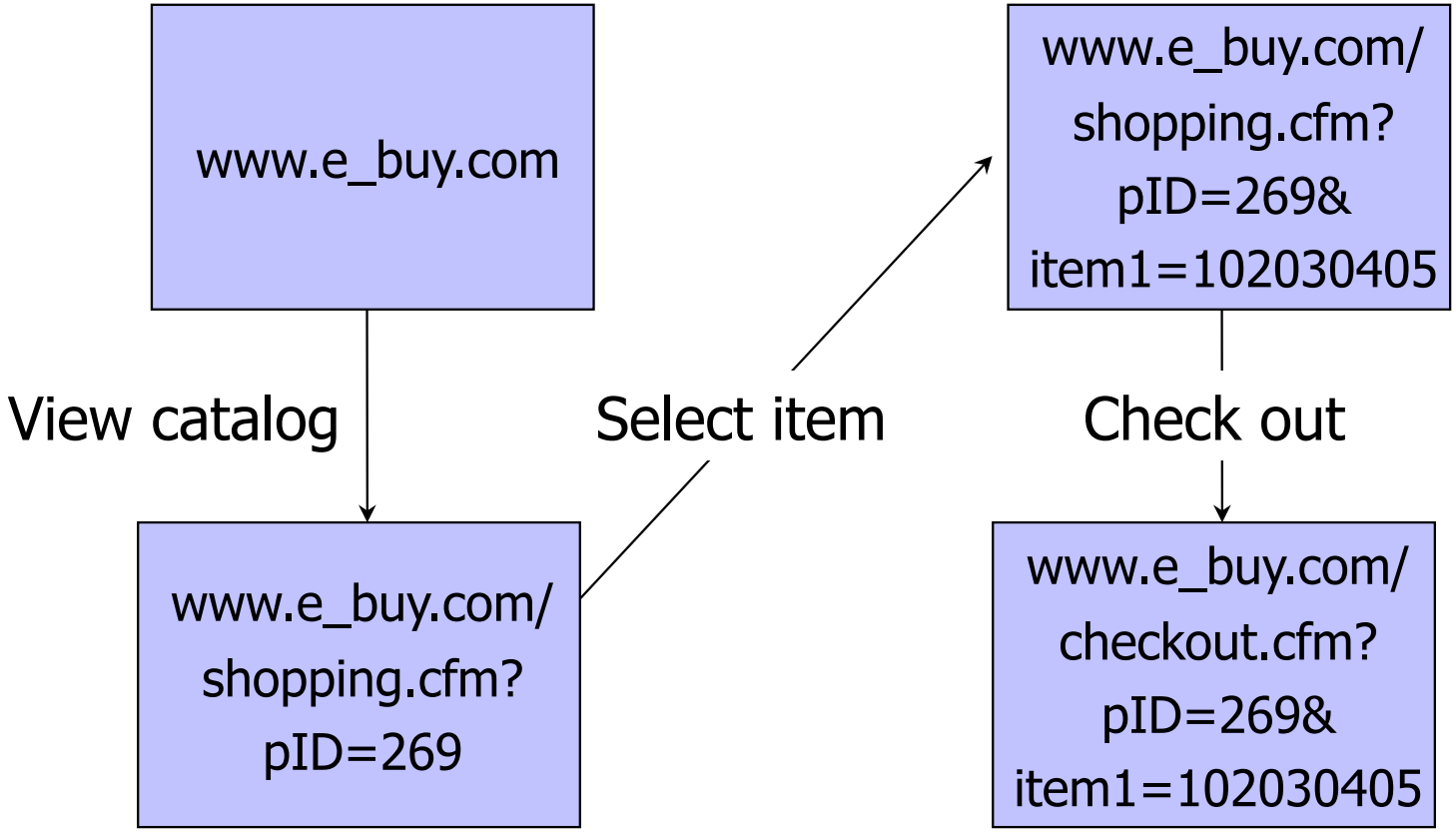
HTTP Request



HTTP Response



Primitive Browser Session



Store session information in URL; easily read on network

FatBrain.com circa 1999 [due to Fu et al.]

- User logs into website with his password, authenticator is generated, user is given special URL containing the authenticator

  https://www.fatbrain.com/HelpAccount.asp?t=0&p1=me@me.com&p2=540555758

  - With special URL, user doesn't need to re-authenticate
    - Reasoning: user could not have not known the special URL without authenticating first. That's true, BUT…
- Authenticators are global sequence numbers
  - It's easy to guess sequence number for another user

    https://www.fatbrain.com/HelpAccount.asp?t=0&p1=SomeoneElse&p2=540555752
  - Partial fix: use random authenticators
    - (Why not complete fix?)

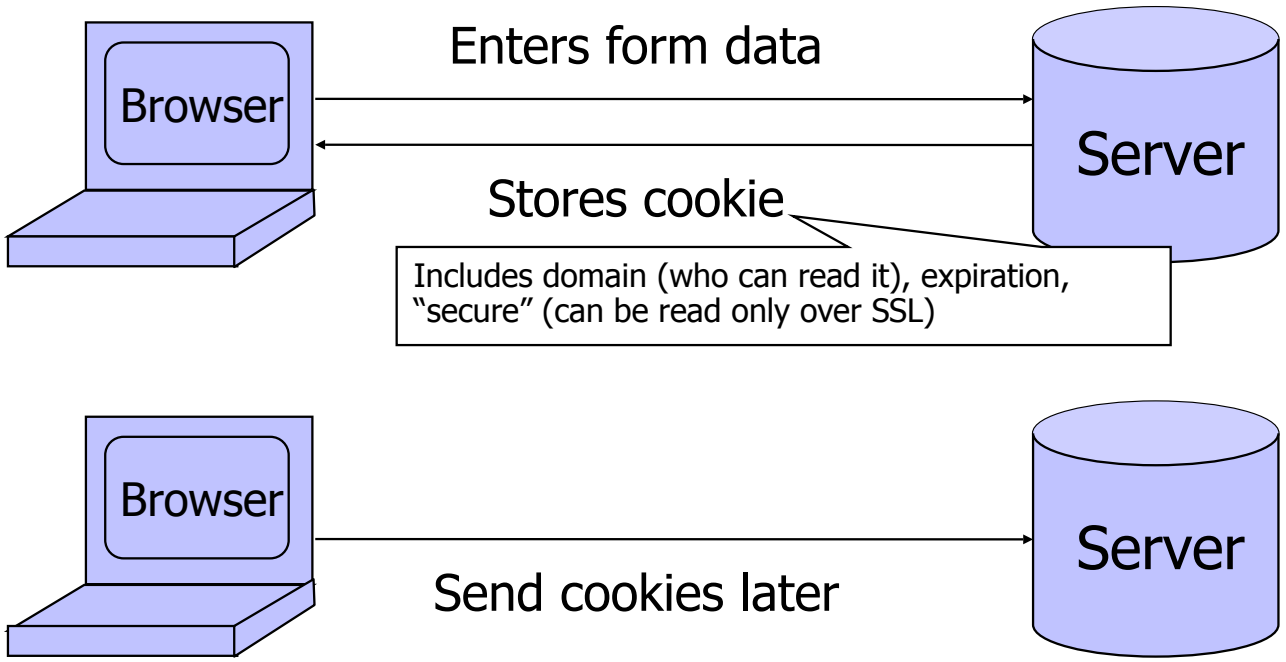
Bad Idea: Encoding State in URL

- Unstable, frequently changing URLs
- Vulnerable to eavesdropping
- There is no guarantee that URL is private
  - Early versions of Opera used to send entire browsing history, including all visited URLs, to Google

Cookies

Storing Info Across Sessions

- A cookie is a file created by an Internet site to store information on your computer



HTTP is a stateless protocol; cookies add state

What Are Cookies Used For?

- Authentication
  - Use the fact that the user authenticated correctly in the past to make future authentication quicker
- Personalization
  - Recognize the user from a previous visit
- Tracking
  - Follow the user from site to site; learn his/her browsing behavior, preferences, and so on

Cookie Management

- Cookie ownership
  - Once a cookie is saved on your computer, only the website that created the cookie can read it (supposedly)
- Variations
  - Temporary cookies
    - Stored until you quit your browser
  - Persistent cookies
    - Remain until deleted or expire
  - Third-party cookies
    - Originates on or sent to another website

Privacy Issues with Cookies

- Cookie may include any information about you known by the website that created it
  - Browsing activity, account information, etc.
- Sites can share this information
  - Advertising networks
  - 2o7.net tracking cookie
- Browser attacks could invade your privacy

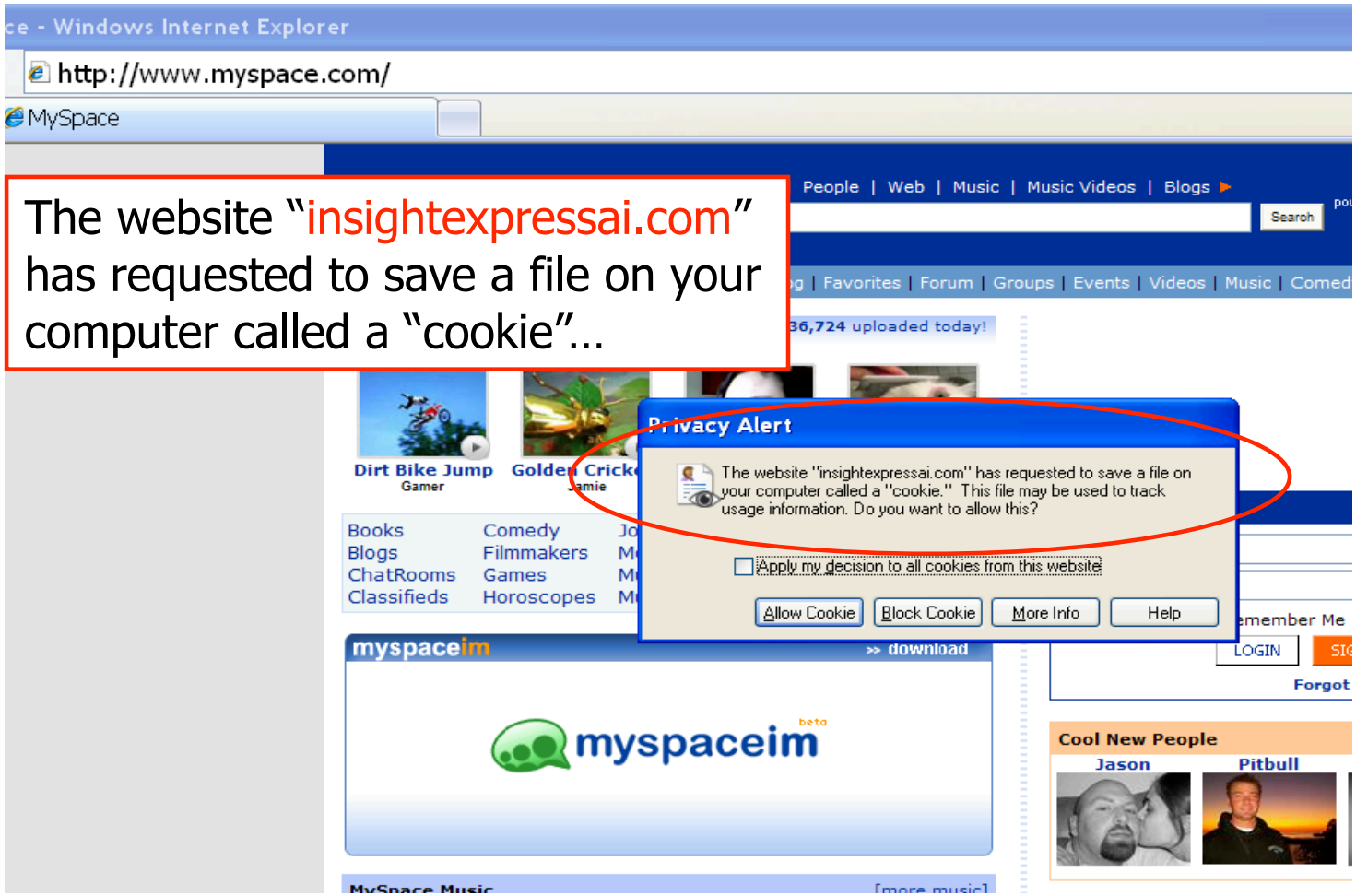
  November 8, 2001:

  Users of Microsoft's browser and e-mail programs could be vulnerable to having their browser cookies stolen or modified due to a new security bug in Internet Explorer (IE), the company warned today

The Weather Channel

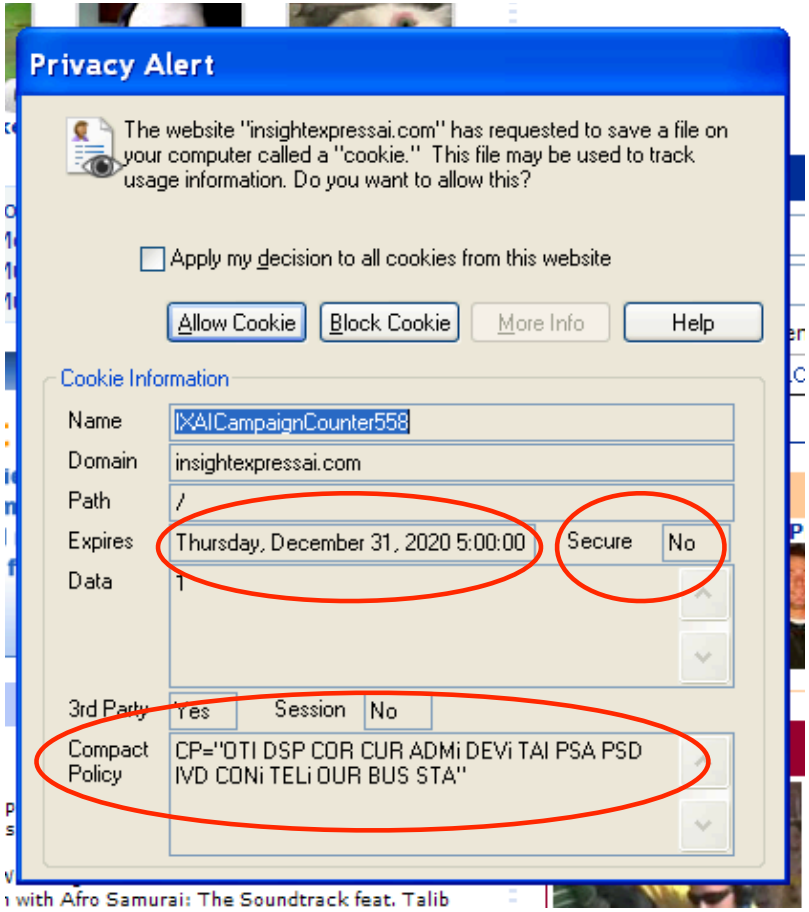
The website "twci.coremetrics.com" has requested to save a file on your computer called a "cookie." This file may be used to track usage information…

MySpace



The website "insightexpressai.com" has requested to save a file on your computer called a "cookie"…

Let's Take a Closer Look…



Storing State in Browser

- Dansie Shopping Cart (2006)
  - "A premium, comprehensive, Perl shopping cart. Increase your web sales by making it easier for your web store customers to order."

```
<FORM METHOD=POST
 ACTION="http://www.dansie.net/cgi-bin/scripts/cart.pl">

  Black Leather purse with leather straps<BR>Pri

  <INPUT TYPE=HIDDEN NAME=name     VALUE="Black leather purse">
  <INPUT TYPE=HIDDEN NAME=price    VALUE="20.00">
  <INPUT TYPE=HIDDEN NAME=sh       VALUE="1">
  <INPUT TYPE=HIDDEN NAME=img      VALUE="purse.jpg">
  <INPUT TYPE=HIDDEN NAME=custom1  VALUE="Black leather purse with
leather straps">

  <INPUT TYPE=SUBMIT NAME="add" VALUE="Put in Shopping Cart">

</FORM>
```

Change this to 2.00

Bargain shopping!

Shopping Cart Form Tampering

http://xforce.iss.net/xforce/xfdb/4621

- Many Web-based shopping cart applications use hidden fields in HTML forms to hold parameters for items in an online store. These parameters can include the item's name, weight, quantity, product ID, and price. Any application that bases price on a hidden field in an HTML form is vulnerable to price changing by a remote user. A remote user can change the price of a particular item they intend to buy, by changing the value for the hidden HTML tag that specifies the price, to purchase products at any price they choose.
- **Platforms Affected:**
  - 3D3.COM Pty Ltd: ShopFactory 5.8 and earlier
  - @Retail Corporation: @Retail Any version
  - Adgrafix: Check It Out Any version
  - Baron Consulting Group: WebSite Tool Any version
  - ComCity Corporation: SalesCart Any version
  - Crested Butte Software: EasyCart Any version
  - Dansie.net: Dansie Shopping Cart Any version
  - Intelligent Vending Systems: Intellivend Any version
  - Make-a-Store: Make-a-Store OrderPage Any version
  - McMurtrey/Whitaker & Associates: Cart32 2.6
  - McMurtrey/Whitaker & Associates: Cart32 3.0
  - pknutsen@nethut.no: CartMan 1.04
  - Rich Media Technologies: JustAddCommerce 5.0
  - SmartCart: SmartCart Any version
  - Web Express: Shoptron 1.2

Storing State in Browser Cookies

- Set-cookie: price=299.99
- User edits the cookie… cookie: price=29.99
- What's the solution?
- Add a MAC to every cookie, computed with the server's secret key
  - Price=299.99; MAC(ServerKey, 299.99)
- Is this the solution?

Storing State in Browser

- Dansie Shopping Cart (2006)
  - "A premium, comprehensive, Perl shopping cart. Increase your web sales by making it easier for your web store customers to order."

```
<FORM METHOD=POST
 ACTION="http://www.dansie.net/cgi-bin/scripts/cart.pl">

  Black Leather purse with leather straps<BR>Price: $20.00<BR>

  <INPUT TYPE=HIDDEN NAME=name     VALUE="Black leather purse">
  <INPUT TYPE=HIDDEN NAME=price    VALUE="F13A3....B2"> A319F...3C
  <INPUT TYPE=HIDDEN NAME=sh       VALUE="1">
  <INPUT TYPE=HIDDEN NAME=img      VALUE="purse.jpg">
  <INPUT TYPE=HIDDEN NAME=custom1  VALUE="Black leather purse      with
leather straps">

  <INPUT TYPE=SUBMIT NAME="add" VALUE="Put in Shopping Cart">

</FORM>
```

MAC(K, "$20")

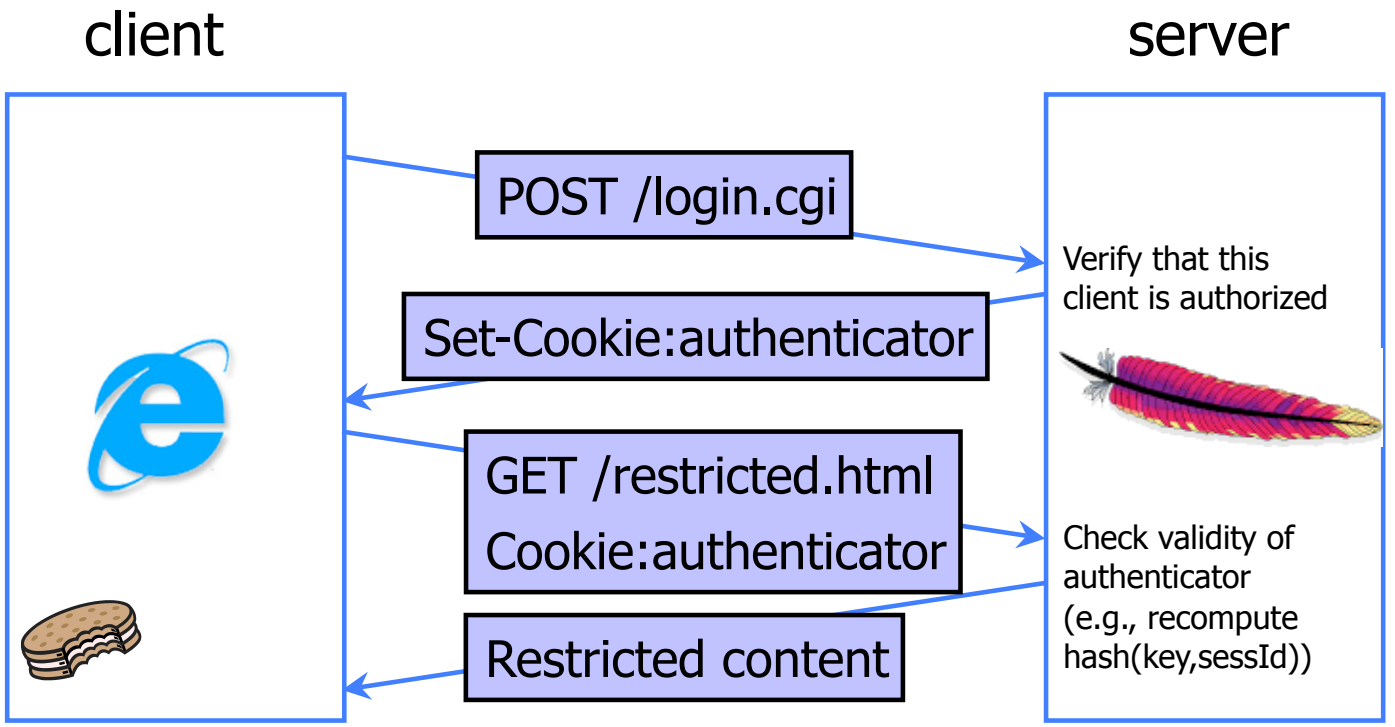
MAC(K, "$2")

Better: MAC(K, "$20,Black leather purse, product number 12345, ...")

Web Authentication via Cookies

- Need authentication system that works over HTTP and does not require servers to store session data
  - Why is it a bad idea to store session state on server?
- Servers can use cookies to store state on client
  - When session starts, server computes an authenticator and gives it back to browser in the form of a cookie
    - Authenticator is a value that client cannot forge on his own
    - Example: MAC(server's secret key, session id)
  - With each request, browser presents the cookie
  - Server recomputes and verifies the authenticator
    - Server does not need to remember the authenticator

Typical Session with Cookies



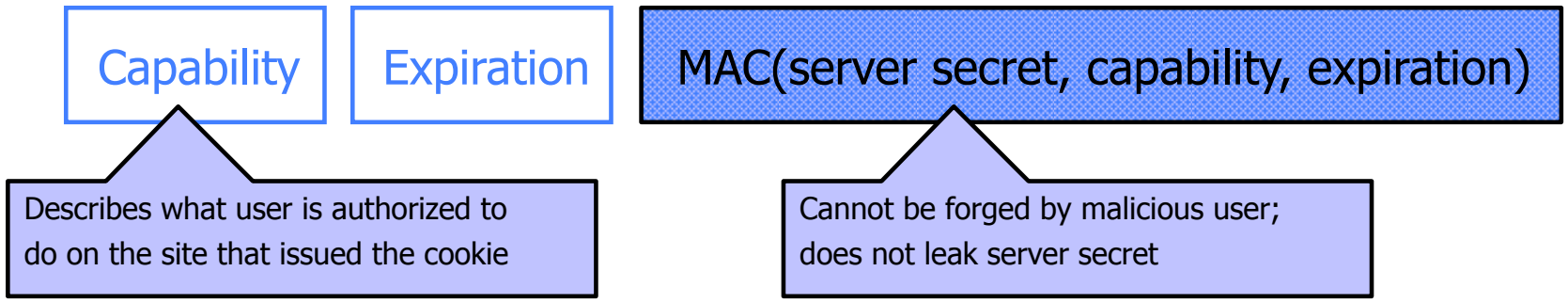
Authenticators must be unforgeable and tamper-proof

(malicious client shouldn't be able to compute his own or modify an existing authenticator)

WSJ.com circa 1999 [due to Fu et al.]

- Idea: use user,hash(user||key) as authenticator
  - Key is secret and known only to the server. Without the key, clients can't forge authenticators.
  - || is string concatenation
- Implementation: user,crypt(user||key)
  - crypt() is UNIX hash function for passwords
  - crypt() truncates its input at 8 characters
  - Usernames matching first 8 characters end up with the same authenticator
  - No expiration or revocation
- It gets worse… This scheme can be exploited to extract the server's secret key

Better Cookie Authenticator



- Main lesson: don't roll your own!
  - Homebrewed authentication schemes are often flawed
- There are standard cookie-based schemes